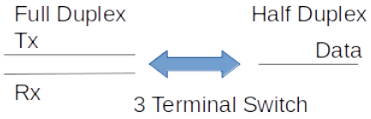
Controlling Dynamixel XL-320 with Arduino

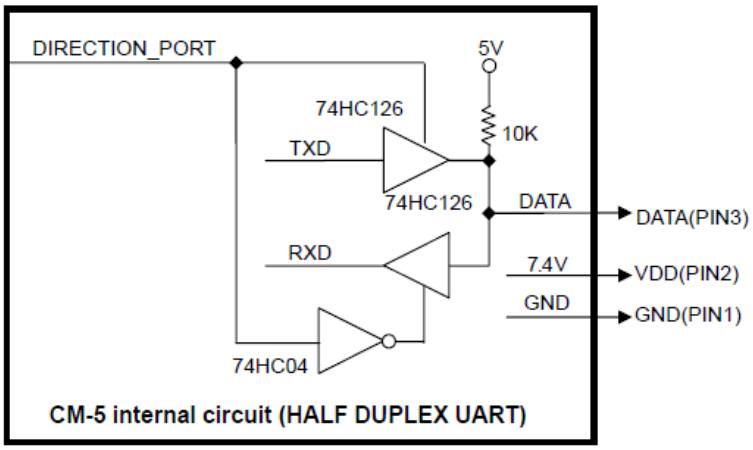
January 25,2020.

Introduction

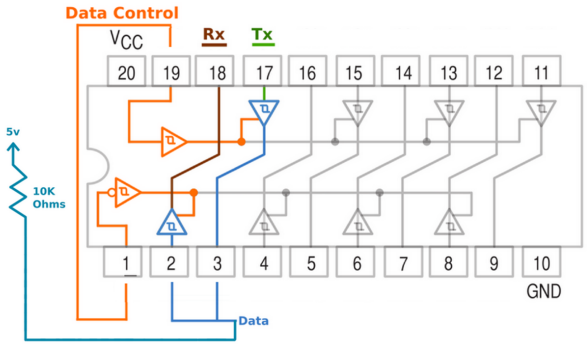
The Dynamixel XL-320 servo motor has the ability to work in continuous mode, servo mode, PID control mode and provides velocity, position, temperature feedback among other things. It uses Half duplex asynchronous serial communication, micro-controllers like Arduino, Raspberry use Full duplex asynchronous serial communication so it is necessary to use external circuitry to make communication possible between the Arduino and XL-320.



Internal circuit of Half duplex :



Hardware setup



This three terminal switch is realised with a tri state buffer, 74LS241. The Data control pin is tied to one of the digital pins on the Arduino. With the data control pin set to high value, Transmission channel turns on and commands can be sent to the motor. Setting the data control pin to low value turns on Receiver channel thereby allowing feedback. This will allow us to have full communication with the XL-320. XL-320 has a response delay of 0.5ms(default value). The default value should be fine, but if it is set too low, then the Arduino might not be able to switch the tri-state buffer in time to receive the return message from the XL320.

We chose Arduino Due as it provides four Serial ports. We require two ports, one for the XL-320 and the other for Serial monitor on the Computer.

DC power supply supplies a voltage of 7.4V. 74LS241 requires 5V and XL-320 requires 7.4V so 74LS241 and XL-320 are powered seperately. Arduino has on-board voltage regulator which can be accessed with the Vin pin. 7.4V is stepped down to 5V by powering the Vin pin to 7.4V. Now connect the 74LS241 to the generated 5V, XL-320 is powered by the DC power supply.

Code

Install the Dynamixel2Arduino library(can be found in Boards Manager tab) on the Arduino IDE. The readwrite_controltable sample code available in the library can be edited to write and read all the data from EEPROM and RAM of the Servo's Memory Area.